

Plan-types de documents

Bruno Bouzy

Juin 2001

5. ANNEXES - PLANS TYPE DE DOCUMENT

Dans ce paragraphe, sont réunis des plans type de certains documents cités au §3.

Document de spécifications des besoins du logiciel

1. INTRODUCTION

1.1. But du document

Ce paragraphe est très court. Il s'agit là de présenter la phase de spécifications des besoins du logiciel d'un point de vue méthodologique. Il dépend très peu de l'application elle-même.

1.2. Contexte de l'application

Ce paragraphe présente l'application dans son contexte.

1.3. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. SPECIFICATIONS GENERALES

2.1. Présentation du logiciel

2.1.1. Environnement du logiciel

Ce paragraphe décrit sommairement les logiciels environnants.

2.1.2. Description générale du logiciel

Ce paragraphe décrit en quelques lignes les fonctionnalités du logiciel. Il s'agit de donner un premier aperçu sans rentrer dans le détail.

2.2. Contraintes opérationnelles

2.2.1. Exploitation

Ce paragraphe précise les contraintes rencontrées en mode opérationnel s'il peut en exister de différentes de celles rencontrées en mode de développement.

2.2.2. Performances

Ce paragraphe précise les temps de réponses attendus.

2.2.3. Installation

Ce paragraphe précise sous quelle forme est livré le logiciel: source et/ou exécutable, éventuellement méthodes de création d'exécutables à partir des sources.

2.3. Contraintes de réalisation

2.3.1. Contraintes structurelles

Ce paragraphe décrit les choix déjà faits ou obligés qui pourront affecter la structure du logiciel.

2.3.2. Contraintes d'évolution

Ce paragraphe précise si le logiciel est développé en prévoyant de le faire évoluer pour le réutiliser dans des applications similaires.

2.3.3. Contraintes de développement

Ce paragraphe décrit le matériel, le langage, les outils et les méthodes de développement.

2.3.4. Contraintes de qualité

Ce paragraphe liste les critères de qualité retenus pour déterminer si le logiciel est de qualité.

3. SPECIFICATIONS FONCTIONNELLES ET/OU OBJET

Ce paragraphe peut être élaboré suivant deux plans différents selon la méthodologie en vigueur : spécifications fonctionnelles ou spécifications "objet". Ce paragraphe constitue le réel contenu de ce document. On peut aussi - et c'est conseillé - donner les deux types de spécification (fonctionnelle et objet). Cela permet d'être plus complet en approchant les besoins de l'utilisateur suivant deux points de vue complémentaires.

3. SPECIFICATIONS OBJET

Ce paragraphe peut être élaboré avec une méthode de spécification orientée objet. On pourra par exemple s'appuyer sur le formalisme OMT ou UML. On élaborera plusieurs "feuilletts" avec des diagrammes de classe. (cf formalisme graphique OMT UML). On pourra éventuellement compléter ces diagrammes par une liste de descriptions textuelles des classes (cf ci-dessous).

3.x Classe x

3.x.1 Description générale de la classe

3.x.2 Super-classe et sous-classe

La classe mère et les classes fille de la classe

3.x.2 Attributs

Les attributs publics accessibles par l'utilisateur du logiciel

3.x.2 Associations

Les associations avec d'autres classes

3.x.3 Opérations

Les méthodes publiques accessibles par l'utilisateur du logiciel

3. SPECIFICATIONS FONCTIONNELLES

Ce paragraphe peut être élaboré avec l'aide d'une méthode de spécification de logiciel. L'exemple de plan qui suit s'appuie sur les principes de décomposition fonctionnelle de SADT.

3.x. Fonction x

3.x.1 Présentation de la fonction

Ce paragraphe présente la fonction d'un point de vue synthétique.

3.x.2 Entrées

Ce paragraphe liste les informations en entrées de la fonction.

3.x.3 Traitements

Ce paragraphe liste les traitements effectués.

3.x.4 Sorties

Ce paragraphe liste les informations produites par la fonction.

4. SPECIFICATIONS D'INTERFACE

Les interfaces du logiciel peuvent être de plusieurs types : Homme/Logiciel, Logiciel/Logiciel, Logiciel/Matériel. Ce paragraphe permet de les spécifier.

4.1. Interface Homme/Logiciel

Interface x

Le plan de cette partie n'est pas élaboré

4.2. Interfaces Logiciel/Logiciel

Les interfaces Logiciel/Logiciel peuvent être de deux types:

- relation d'utilisation entre deux modules logiciels,*
- communication entre deux logiciels qui s'exécutent en parallèle*

4.2.1. Interfaces d'utilisation

4.2.1.x. Interface x

Ce paragraphe décrit les primitives de l'interface avec leur type, leurs arguments, leur valeur retournée, leur conditions d'utilisation.

4.2.2. Interfaces de communication

4.2.2.x. Interface x

4.2.2.x.1. But et informations échangées

Ce paragraphe décrit le but de l'échange d'informations, la signification, le sens, la taille et la fréquence des informations échangées.

4.2.2.x.2. Procédures d'échange

Ce paragraphe décrit la manière dont on va échanger les informations (synchronisation, protocole, format des données, condition de déclenchement, temps de réponse).

4.3. Interface Logiciel/Matériel

Ce paragraphe décrit les instructions du matériel utilisés si cela est nécessaire, les signaux envoyés par le matériel, les retour d'erreurs.

On peut aussi utiliser les cas d'utilisations de UML pour spécifier cette partie. Un plan possible est alors le suivant :

4.1. Description des acteurs

4.1.1. acteurs principaux

4.1.2. acteurs secondaires

4.1.3. matériel externe

4.1.4. autres systèmes

4.2. Description des cas

4.2.x cas X

Plan de développement du logiciel

1. INTRODUCTION

1.1. But du document

Ce paragraphe permet de présenter le développement du logiciel dans son ensemble.

1.2. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. ORGANISATION

Ce paragraphe décrit l'ensemble des activités introduites dans l'organigramme des tâches de la gestion de projet.

2.1. Décomposition en tâches

2.2. Structure des équipes

3. PLANIFICATION

Ce paragraphe décrit l'enchaînement des tâches au cours du développement du projet. Les dates de fin de phases de développement et de remises de fournitures.

4. CYCLE DE VIE

Ici sont présentés les généralités de la méthode de développement utilisée avec la décomposition du développement en phases successives. Cf plus haut.

5. METHODES ET OUTILS

Description technique des outils du développement.

6. DOCUMENTATION

6.1. Présentation

Ici sont présentés les documents fournis phase par phase avec leur plan-type.

6.2. Standard et outils

Ici sont cités les outils et standard qui supporteront la documentation.

7. QUALITE

Ici sont cités les critères qui détermineront que le logiciel est de qualité ou non.

Cahier de recette du logiciel

1. INTRODUCTION

1.1. But du document

1.2. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. DESCRIPTION DE LA FOURNITURE

Ce paragraphe décrit sous quelle forme sera livré le logiciel.

3. MOYENS D'ESSAI ET OUTILS

Ce paragraphe décrit les moyens et les outils permettant de vérifier la conformité du logiciel avec les spécifications des besoins.

4. CONFORMITE AUX SPECIFICATIONS GENERALES

5. CONFORMITE AUX SPECIFICATIONS FONCTIONNELLES OU OBJET

Ce paragraphe décrit les essais qui permettront d'accepter le logiciel du point de vue fonctionnel.

5.x. Essai x

5.x.1. Description

Ce paragraphe décrit le but et le principe de réalisation de l'essai x.

5.x.2. Jeux d'essais

Ce paragraphe décrit les données en entrée, le résultats attendus et les critères d'acceptation de l'essai x.

6. CONFORMITE AUX SPECIFICATIONS D'INTERFACES

Ce paragraphe décrit les essais qui permettront d'accepter le logiciel du point de vue de ses interfaces.

7. CONFORMITE DE LA DOCUMENTATION

Document de conception générale

1. INTRODUCTION

1.1. But du document

Ce paragraphe est très court. Il s'agit là de présenter la phase de conception du logiciel d'un point de vue méthodologique. Il dépend très peu de l'application elle-même.

1.2. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. ARCHITECTURE DYNAMIQUE DU LOGICIEL

Ce paragraphe décrit l'organisation générale des tâches du logiciel. Un ou plusieurs schémas représentant cette organisation peut être nécessaire.

2.x. Tâche x

2.x.1. Références au SBL

Ce paragraphe fait référence aux fonctions du document de spécifications des besoins du logiciel qui sont implémentées par la tâche x.

2.x.2. Traitements effectués

Ce paragraphe décrit les traitements effectués par la tâche x.

2.x.3. Enchaînements et flots de données

Ce paragraphe décrit les flots de données de la tâche x en entrées et en sorties. La description reste générale.

2.x.4. Modules utilisés

Ce paragraphe donne la liste des modules utilisés pour l'exécution de la tâche x.

La conception générale du logiciel n'étant généralement pas faite du premier jet, on peut très bien rajouter des paragraphes expliquant les autres possibilités de conception avec leurs avantages et leurs inconvénients.

3. DECOUPAGE DU LOGICIEL EN MODULES OU EN CLASSES

Ce paragraphe décrit le découpage en modules ou en classes selon que l'on utilise une méthode de conception orientée objet ou pas. Si l'on utilise une méthode de conception orientée objet, les paragraphes seront des classes, sinon ils seront des modules. Si l'on a utilisé une méthode de spécification orientée objet dans la phase précédente, on utilise sans doute une méthode de conception orientée objet et dans ce cas, le modèle objet présenté dans cette partie affine le modèle objet présenté dans le document de la phase précédente. Cet affinement se traduit en majorité par une extension des classes, par des remplacements ou des ajouts de classe.

3.x. Module x

Ce paragraphe décrit le module de façon synthétique: ensemble des procédures et données regroupées dans le module.

3.x Classe x

Ce paragraphe décrit les classes. Pour chaque classe, on précise ses attributs, ses opérations, ses associations, sa super-classe et ses sous-classes non plus en termes de l'utilisateur extérieur du logiciel mais en termes de l'utilisateur de la classe. La description externe de la classe peut être incomplète (en pratique elle l'est toujours). La description externe complète sera faite dans la phase suivante.

3.x.1 Description générale de la classe

3.x.2 Super-classe et sous-classe

La classe mère et les classes fille de la classe

3.x.2 Attributs

Les attributs publics accessibles par les classes utilisatrices

3.x.2 Associations

Les associations avec d'autres classes

3.x.3 Opérations

Les méthodes publiques accessibles par les classes utilisatrices

4. PARAMETRES

Ce paragraphe décrit les paramètres du système (données de configuration concernant le site, les variables du système modifiables).

Document de conception détaillée

1. INTRODUCTION

1.1. But du document

Ce paragraphe est très court. Il s'agit là de présenter la phase de conception détaillée du logiciel d'un point de vue méthodologique. Il dépend très peu de l'application elle-même.

1.2. Documents de référence

Ce paragraphe liste tous les documents qui ont servi à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. INTERFACES ENTRE LES TACHES DU LOGICIEL

2.x. tâche x

2.x.1. Enchainements et flots de données

Ce paragraphe décrit les flots de données de la tâche x en entrées et en sorties de manière précise (synchronisation, protocole, format et type des données).

3. DESCRIPTION DES MODULES OU DES CLASSES

Ce paragraphe permet de décrire les interfaces des modules, c'est-à-dire les définitions de types, les procédures externes avec les paramètres d'entrée et de sortie et leur valeur retournée, les données modulaires externes et les relations d'utilisation des autres modules. Dans le cas d'une conception orientée objet un paragraphe ne sera pas un module mais la description externe (ou interface) d'une classe d'objets.

3.x. Module x

3.x.1. Rappel des objectifs du module

Ce paragraphe peut reprendre la description du logiciel faite lors de la conception générale.

3.x.2. Relations d'utilisation avec d'autres modules

Ce paragraphe donne la liste des modules utilisés par ce module et la liste des modules qui utilisent ce module.

3.x.3. Définitions de types

Ce paragraphe donne la liste des définitions de type. Dans le cas d'une conception orientée objets cette définition de type peut être unique et représenter les attributs de l'objet associé au module.

3.x.4. Procédures externes

Ce paragraphe donne la liste des procédures visibles par les modules qui utilisent ce module. Dans le cas d'une conception orientée objets ces procédures externes peuvent être les méthodes associées à l'objet qui est associé au module.

3.x.5. Variables externes

Ce paragraphe donne la liste des variables visibles par les modules qui utilisent ce module.

3.x Classe x

Ce paragraphe décrit l'extérieur de la classe avec ses attributs, opérations, associations, super-classe et sous-classes. La description doit être complète. Elle servira de contrat pour la phase de codage lorsque l'intérieur de la classe sera implémenté.

3.x.1 Description complète de la classe

3.x.2 Super-classe et sous-classe

La classe mère et les classes fille de la classe

3.x.2 Attributs

Les attributs publics accessibles par l'utilisateur du logiciel

3.x.2 Associations

Les associations avec d'autres classes

3.x.3 Opérations

Les méthodes publiques accessibles par l'utilisateur du logiciel

Document de spécification des tests d'intégration

1. INTRODUCTION

1.1. But du document

Ce paragraphe permet de présenter le but de la spécifications des tests d'intégration.

1.2. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. DESCRIPTION DES TESTS D'INTEGRATION

2.1.x. Test x

2.1.x.1. Description du test

Ce paragraphe décrit le test de façon externe.

2.1.x.2. But du test

Ce paragraphe décrit ce que prouve le test.

2.1.x.3. Principe de réalisation

Ce paragraphe décrit comment réaliser le test et liste les jeux d'essais à faire passer.

2.1.x.4. Jeux d'essais

2.1.x.4.1. données en entrées

Ce paragraphe décrit les conditions de départ du test.

2.1.x.4.2. résultats attendus

Ce paragraphe décrit les résultats attendus du test.

2.1.x.4.3. critères d'acceptation du test

Ce paragraphe décrit dans quelle mesure le test sera accepté.

2.2. ORDONNANCEMENT DES TESTS

Ce paragraphe décrit l'ordre dans lequel les tests doivent être exécuté si celui-ci a une importance.

Manuel d'utilisation du logiciel

1. INTRODUCTION

1.1. But du document

Ce paragraphe est très court. Il s'agit là de présenter la phase de conception détaillée du logiciel d'un point de vue méthodologique. Il dépend très peu de l'application elle-même.

1.2. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document. Il peut faire foi en cas de litige entre un client et un fournisseur.

2. MISE EN OEUVRE

2.1. Environnement

Ce paragraphe décrit les conditions de départ qui doivent être réunies avant la mise en oeuvre du système (configuration matériel/logiciel).

2.2. Installation

Ce paragraphe décrit la procédure de lancement.

3. MODE D'EMPLOI

3.x. Fonction x

3.x.1. Entrées

Ce paragraphe décrit les valeurs, le format, les unités de toutes les variables d'entrée de la fonction.

3.x.2. Procédures d'exécution

Ce paragraphe décrit comment lancer la fonction.

3.x.3. Sorties

Ce paragraphe décrit les valeurs, le format, les unités de toutes les variables de sortie de la fonction.

3.x.4. Relations entre fonctions

Ce paragraphe décrit le passage d'une fonction à l'autre.

4. DETECTION D'ERREUR

Ce paragraphe donne la liste des erreurs possibles avec leur description et les procédures de reprises associées.

Résultats des tests unitaires

1. INTRODUCTION

1.1. But du document

1.2. Documents de référence

2. TESTS UNITAIRES

2.x. Tests des procédures du module x

2.x.y. Tests de la procédure y

2.x.y.1. Mise en oeuvre

2.x.y.2. Essais

On précise les caractéristiques de l'essai, la configuration, les résultats obtenus

Résultats des tests d'intégration

1. INTRODUCTION

1.1. But du document

1.2. Documents de référence

2. CONFORMITE AUX SPECIFICATIONS DES TESTS D'INTEGRATION

2.x. Tests x

2.x.1. Mise en oeuvre

2.x.1. Essais

On précise les caractéristiques de l'essai, la configuration, les résultats obtenus

Résultat de la recette du logiciel

1. INTRODUCTION

1.1. But du document

1.2. Documents de référence

Ce paragraphe liste tous les documents qui servent à l'élaboration du présent document.

2. CONFORMITE AUX SPECIFICATIONS GENERALES

3. CONFORMITE AUX SPECIFICATIONS FONCTIONNELLES OU OBJET

3.x. Essai x

3.x.1. Description

Ce paragraphe rappelle le but et le principe de réalisation de l'essai x.

3.x.2. Résultats des jeux d'essais

Ce paragraphe décrit les résultats de l'essai x.

3. CONFORMITE AUX SPECIFICATIONS D'INTERFACES

4. CONFORMITE DE LA DOCUMENTATION